Summary
15/01/2017, 11:35:53

Differences exist between documents.

New Document:	Old Document:
pins_RAMPS or	pins_RAMPS
8 pages (43 KB)	8 pages (43 KB)
15/01/2017, 11:35:51	15/01/2017, 11:35:51
Used to display results.	

Get started: first change is on page 4.

No pages were deleted

How to read this report

Highlight indicates a change.
~~**Deleted**~~ indicates deleted content.
 indicates pages were changed.
 indicates pages were moved.

```
/**
 * Marlin 3D Printer Firmware
 * Copyright (C) 2016 MarlinFirmware [https://github.com/
MarlinFirmware/Marlin]
 *
 * Based on Sprinter and grbl.
 * Copyright (C) 2011 Camiel Gubbels / Erik van der Zalm
 *
 * This program is free software: you can redistribute it and/or
modify
 * it under the terms of the GNU General Public License as published
by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program.  If not, see <http://www.gnu.org/
licenses/>.
 *
 */

/**
 * Arduino Mega with RAMPS v1.4 (or v1.3) pin assignments
 *
 * Applies to the following boards:
 *
 *  RAMPS_14_EFB (Hotend, Fan, Bed)
 *  RAMPS_14_EEB (Hotend0, Hotend1, Bed)
 *  RAMPS_14_EFF (Hotend, Fan0, Fan1)
 *  RAMPS_14_EEF (Hotend0, Hotend1, Fan)
 *  RAMPS_14_SF  (Spindle, Controller Fan)
 *
 *  RAMPS_13_EFB (Hotend, Fan, Bed)
 *  RAMPS_13_EEB (Hotend0, Hotend1, Bed)
 *  RAMPS_13_EFF (Hotend, Fan0, Fan1)
 *  RAMPS_13_EEF (Hotend0, Hotend1, Fan)
 *  RAMPS_13_SF  (Spindle, Controller Fan)
 *
 *  Other pins_MYBOARD.h files may override these defaults
 *
 *  Differences between
 *  RAMPS_13 | RAMPS_14
 *         7 | 11
 */

#if !defined(__AVR_ATmega1280__) && !defined(__AVR_ATmega2560__)
  #error "Oops!  Make sure you have 'Arduino Mega' selected from the
'Tools -> Boards' menu."
#endif
```

```
#ifndef BOARD_NAME
  #define BOARD_NAME "RAMPS 1.4"
#endif

#define LARGE_FLASH true

//
// Servos
//
#ifdef IS_RAMPS_13
  #define SERVO0_PIN        7 // RAMPS_13 // Will conflict with
BTN_EN2 on LCD_I2C_VIKI
#else
  #define SERVO0_PIN       11
#endif
#define SERVO1_PIN          6
#define SERVO2_PIN          5
#define SERVO3_PIN          4

//
// Limit Switches
//
#define X_MIN_PIN           3
#ifndef X_MAX_PIN
  #define X_MAX_PIN         2
#endif
#define Y_MIN_PIN          14
#define Y_MAX_PIN          15
#define Z_MIN_PIN          18
#define Z_MAX_PIN          19

//
// Z Probe (when not Z_MIN_PIN)
//
#ifndef Z_MIN_PROBE_PIN
  #define Z_MIN_PROBE_PIN  32
#endif

#define SLED_PIN           -1

//
// Steppers
//
#define X_STEP_PIN         54
#define X_DIR_PIN          55
#define X_ENABLE_PIN       38
#define X_CS_PIN           53

#define Y_STEP_PIN         60
#define Y_DIR_PIN          61
#define Y_ENABLE_PIN       56
#define Y_CS_PIN           49
```

```
#define Z_STEP_PIN         46
#define Z_DIR_PIN          48
#define Z_ENABLE_PIN       62
#define Z_CS_PIN           40

#define E0_STEP_PIN        26
#define E0_DIR_PIN         28
#define E0_ENABLE_PIN      24
#define E0_CS_PIN          42

#define E1_STEP_PIN        36
#define E1_DIR_PIN         34
#define E1_ENABLE_PIN      30
#define E1_CS_PIN          44

//
// Temperature Sensors
//
#define TEMP_0_PIN         13   // Analog Input
#define TEMP_1_PIN         15   // Analog Input
#define TEMP_BED_PIN       14   // Analog Input

// SPI for Max6675 or Max31855 Thermocouple
#if DISABLED(SDSUPPORT)
  #define MAX6675_SS       66 // Do not use pin 53 if there is even
the remote possibility of using Display/SD card
#else
  #define MAX6675_SS       66 // Do not use pin 49 as this is tied
to the switch inside the SD card socket to detect if there is an SD
card present
#endif

//
// Augmentation for auto-assigning RAMPS plugs
//
#if DISABLED(IS_RAMPS_EEB) && DISABLED(IS_RAMPS_EEF) &&
DISABLED(IS_RAMPS_EFB) && DISABLED(IS_RAMPS_EFF) &&
DISABLED(IS_RAMPS_SF) && !PIN_EXISTS(MOSFET_D)
  #if HOTENDS > 1
    #if TEMP_SENSOR_BED
      #define IS_RAMPS_EEB
    #else
      #define IS_RAMPS_EEF
    #endif
  #elif TEMP_SENSOR_BED
    #define IS_RAMPS_EFB
  #else
    #define IS_RAMPS_EFF
  #endif
#endif

//
// Heaters / Fans
//
```

```
#ifndef MOSFET_D_PIN
  #define MOSFET_D_PIN  -1
#endif
#ifndef RAMPS_D8_PIN
  #define RAMPS_D8_PIN   8
#endif
#ifndef RAMPS_D9_PIN
  #define RAMPS_D9_PIN   6
#endif
#ifndef RAMPS_D10_PIN
  #define RAMPS_D10_PIN 10
#endif

#define HEATER_0_PIN     RAMPS_D10_PIN

#if ENABLED(IS_RAMPS_EFB)                       // Hotend, Fan, Bed
  #define FAN_PIN        RAMPS_D9_PIN
  #define HEATER_BED_PIN RAMPS_D8_PIN
#elif ENABLED(IS_RAMPS_EEF)                     // Hotend, Hotend,
Fan
  #define HEATER_1_PIN   RAMPS_D9_PIN
  #define FAN_PIN        RAMPS_D8_PIN
#elif ENABLED(IS_RAMPS_EEB)                     // Hotend, Hotend,
Bed
  #define HEATER_1_PIN   RAMPS_D9_PIN
  #define HEATER_BED_PIN RAMPS_D8_PIN
#elif ENABLED(IS_RAMPS_EFF)                     // Hotend, Fan, Fan
  #define FAN_PIN        RAMPS_D9_PIN
  #define FAN1_PIN       RAMPS_D8_PIN
#elif ENABLED(IS_RAMPS_SF)                      // Spindle, Fan
  #define FAN_PIN        RAMPS_D8_PIN
#else                                           // Non-specific are
"EFB" (i.e., "EFBF" or "EFBE")
  #define FAN_PIN        RAMPS_D9_PIN
  #define HEATER_BED_PIN RAMPS_D8_PIN
  #if HOTENDS == 1
    #define FAN1_PIN     MOSFET_D_PIN
  #else
    #define HEATER_1_PIN MOSFET_D_PIN
  #endif
#endif

#ifndef FAN_PIN
  #define FAN_PIN 4      // IO pin. Buffer needed
#endif

//
// Misc. Functions
//
#define SDSS               53
#define LED_PIN            13

// Use the RAMPS 1.4 Analog input 5 on the AUX2 connector
#define FILWIDTH_PIN        5   // Analog Input
```

```
// define digital pin 4 for the filament runout sensor. Use the
RAMPS 1.4 digital input 4 on the servos connector
#define FIL_RUNOUT_PIN      4

#define PS_ON_PIN          12

//
// LCD / Controller
//
#if ENABLED(ULTRA_LCD)

  #if ENABLED(REPRAPWORLD_GRAPHICAL_LCD)
    #define LCD_PINS_RS     49 // CS chip select /SS chip slave
select
    #define LCD_PINS_ENABLE 51 // SID (MOSI)
    #define LCD_PINS_D4     52 // SCK (CLK) clock
  #elif ENABLED(NEWPANEL) && ENABLED(PANEL_ONE)
    #define LCD_PINS_RS 40
    #define LCD_PINS_ENABLE 42
    #define LCD_PINS_D4 65
    #define LCD_PINS_D5 66
    #define LCD_PINS_D6 44
    #define LCD_PINS_D7 64
  #else
    #define LCD_PINS_RS 16
    #define LCD_PINS_ENABLE 17
    #define LCD_PINS_D4 23
    #define LCD_PINS_D5 25
    #define LCD_PINS_D6 27
    #define LCD_PINS_D7 29
    #if DISABLED(NEWPANEL)
      #define BEEPER_PIN 33
      // Buttons are attached to a shift register
      // Not wired yet
      //#define SHIFT_CLK 38
      //#define SHIFT_LD 42
      //#define SHIFT_OUT 40
      //#define SHIFT_EN 17
    #endif
  #endif

  #if ENABLED(NEWPANEL)

    #if ENABLED(REPRAP_DISCOUNT_SMART_CONTROLLER)
      #define BEEPER_PIN 37

      #define BTN_EN1 31
      #define BTN_EN2 33
      #define BTN_ENC 35

      #define SD_DETECT_PIN 49
      #define KILL_PIN 41
```

```
      #if ENABLED(BQ_LCD_SMART_CONTROLLER)
        #define LCD_BACKLIGHT_PIN 39
      #endif

    #elif ENABLED(REPRAPWORLD_GRAPHICAL_LCD)
      #define BTN_EN1 64
      #define BTN_EN2 59
      #define BTN_ENC 63
      #define SD_DETECT_PIN 42
    #elif ENABLED(LCD_I2C_PANELOLU2)
      #define BTN_EN1 47  // reverse if the encoder turns the wrong
way.
      #define BTN_EN2 43
      #define BTN_ENC 32
      #define LCD_SDSS 53
      #define SD_DETECT_PIN -1
      #define KILL_PIN 41
    #elif ENABLED(LCD_I2C_VIKI)
      #define BTN_EN1 22  // reverse if the encoder turns the wrong
way.
      #define BTN_EN2 7   // http://files.panucatt.com/datasheets/
viki_wiring_diagram.pdf
                          // tells about 40/42.
                          // 22/7 are unused on RAMPS_14. 22 is
unused and 7 the SERVO0_PIN on RAMPS_13.
      #define BTN_ENC -1
      #define LCD_SDSS 53
      #define SD_DETECT_PIN 49
    #elif ENABLED(VIKI2) || ENABLED(miniVIKI)
      #define BEEPER_PIN       33

      // Pins for DOGM SPI LCD Support
      #define DOGLCD_A0        44
      #define DOGLCD_CS        45
      #define LCD_SCREEN_ROT_180

      #define BTN_EN1          22
      #define BTN_EN2           7
      #define BTN_ENC          39

      #define SDSS             53
      #define SD_DETECT_PIN    -1  // Pin 49 for display sd
interface, 72 for easy adapter board

      #define KILL_PIN         31

      #define STAT_LED_RED_PIN 32
      #define STAT_LED_BLUE_PIN 35

    #elif ENABLED(ELB_FULL_GRAPHIC_CONTROLLER)
      #define BTN_EN1 35  // reverse if the encoder turns the wrong
way.
      #define BTN_EN2 37
      #define BTN_ENC 31
```

```
  #define SD_DETECT_PIN 49
  #define LCD_SDSS 53
  #define KILL_PIN 41
  #define BEEPER_PIN 23
  #define DOGLCD_CS 29
  #define DOGLCD_A0 27
  #define LCD_BACKLIGHT_PIN 33
#elif ENABLED(MINIPANEL)
  #define BEEPER_PIN 42
  // Pins for DOGM SPI LCD Support
  #define DOGLCD_A0  44
  #define DOGLCD_CS  66
  #define LCD_BACKLIGHT_PIN 65 // backlight LED on A11/D65
  #define SDSS   53

  #define KILL_PIN 64
  // GLCD features
  //#define LCD_CONTRAST 190
  // Uncomment screen orientation
  //#define LCD_SCREEN_ROT_90
  //#define LCD_SCREEN_ROT_180
  //#define LCD_SCREEN_ROT_270
  // The encoder and click button
  #define BTN_EN1 40
  #define BTN_EN2 63
  #define BTN_ENC 59
  // not connected to a pin
  #define SD_DETECT_PIN 49

#else

  // Beeper on AUX-4
  #define BEEPER_PIN 33

  // buttons are directly attached using AUX-2
  #if ENABLED(REPRAPWORLD_KEYPAD)
    #define BTN_EN1 64 // encoder
    #define BTN_EN2 59 // encoder
    #define BTN_ENC 63 // enter button
    #define SHIFT_OUT 40 // shift register
    #define SHIFT_CLK 44 // shift register
    #define SHIFT_LD 42 // shift register
  #elif ENABLED(PANEL_ONE)
    #define BTN_EN1 59 // AUX2 PIN 3
    #define BTN_EN2 63 // AUX2 PIN 4
    #define BTN_ENC 49 // AUX3 PIN 7
  #else
    #define BTN_EN1 37
    #define BTN_EN2 35
    #define BTN_ENC 31 // the click
  #endif

  #if ENABLED(G3D_PANEL)
    #define SD_DETECT_PIN 49
```

```
        #define KILL_PIN 41
      #else
        //#define SD_DETECT_PIN -1 // Ramps doesn't use this
      #endif

    #endif
  #endif // NEWPANEL

#endif // ULTRA_LCD
```